Java mapping for pragmatic programmers

October 8, 2013

Java mapping for pragmatic programmers

October 8, 2013

Disclaimer

Disclaimer

- I'm not a Scala expert, actually, I do not know Scala at all.

Disclaimer

- I'm not a Scala expert, actually, I do not know Scala at all.
- Lets discuss today not implementations, but ideas staying behind them.

Agenda

1. Parser combinators
2. JAXB plugin
3. Scala Macros
4. Q&A, Discussion

XRools

- Written by Scala experts from Kiev
- XPath like DSL
- Extendable by Scala code
- XML oriented

XRools

Talk is cheap. Show me the code.

XRools Pros

- Scala

XRools Pros

- Scala
- Non-developers write rules

XRools Pros

- Scala
- Non-developers write rules
- Dashboard

XRools Cons

- Scala

XRools Cons

- Scala
- Non-developers write rules

XRools Cons

- Scala
- Non-developers write rules
- No compile-time checks

XRools Cons

- Scala
- Non-developers write rules
- No compile-time checks
- Uses reflection

XRools Cons

- Scala
- Non-developers write rules
- No compile-time checks
- Uses reflection
- No real IDE support

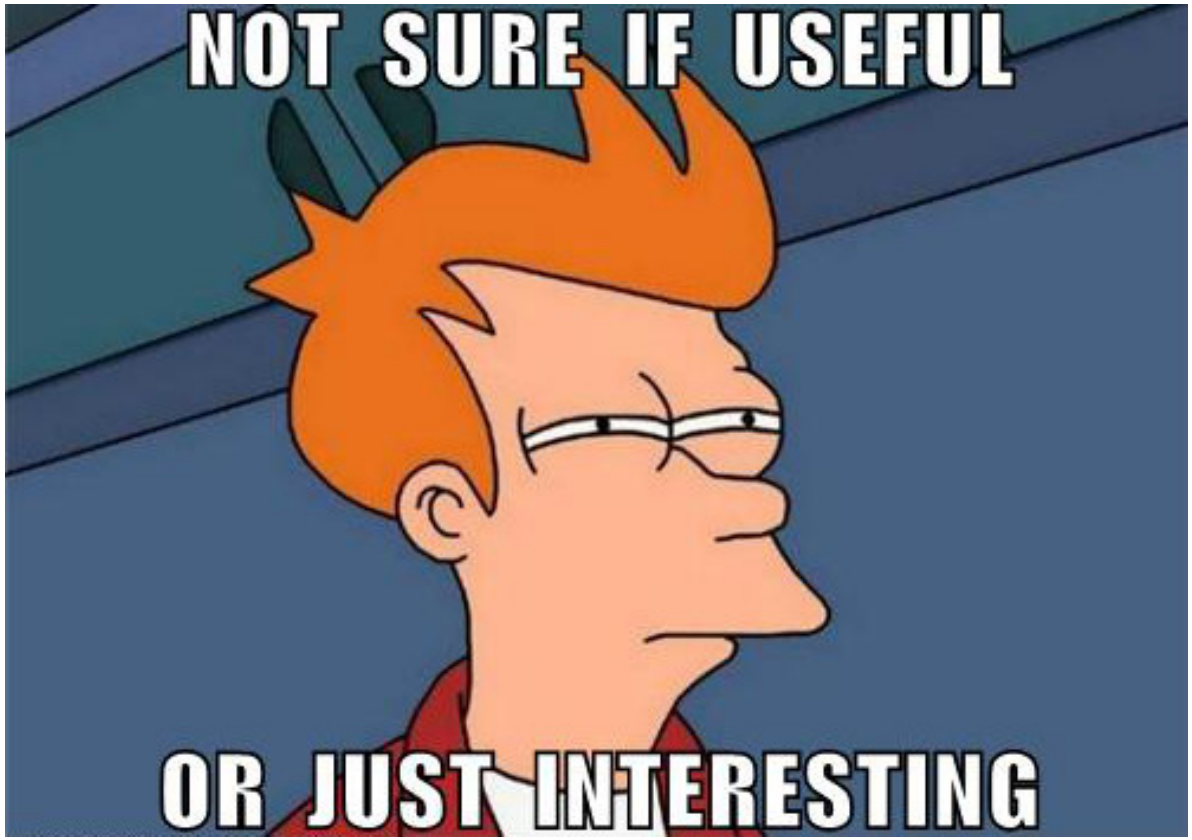
XRools Cons

- Scala
- Non-developers write rules
- No compile-time checks
- Uses reflection
- No real IDE support
- Functions defined away from its only usage

XRools Cons

- Scala
- Non-developers write rules
- No compile-time checks
- Uses reflection
- No real IDE support
- Functions defined away from its only usage
- Debugging?

XRools



JAXB plugin

- Written by Sergey Armensky (idea of Andrey Vytnov?)
- Plain Java (new method added for each property)
- Will be soon in production
- Have nothing to do with Scala

JAXB plugin

Talk is cheap. Show me the code.

JAXB plugin Pros

- Pure Java

JAXB plugin Pros

- Pure Java
- Simple and clean

JAXB plugin Pros

- Pure Java
- Simple and clean
- Simplify some code

JAXB plugin Cons

- Only for JAXB generated classes

JAXB plugin Cons

- Only for JAXB generated classes
- Simplify only some code

JAXB plugin Cons

- Only for JAXB generated classes
- Simplify only some code
- Not declarative enough

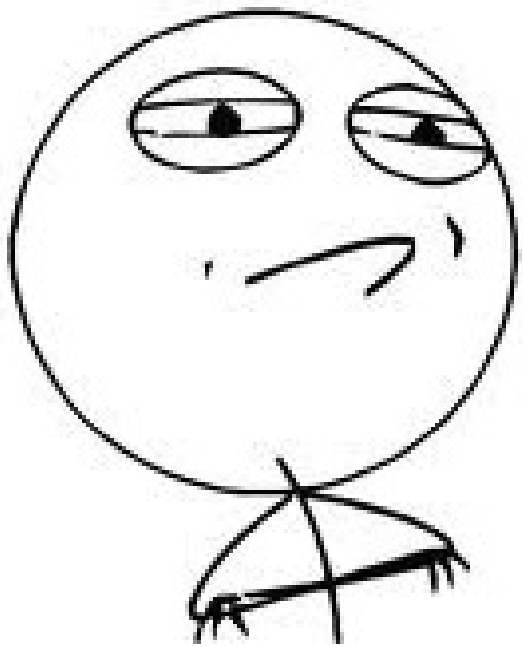
JAXB plugin Cons

- Only for JAXB generated classes
- Simplify only some code
- Not declarative enough
- Not mapper at all

JAXB Plugin

Can we do better?

Can we do better?



CHALLENGE ACCEPTED

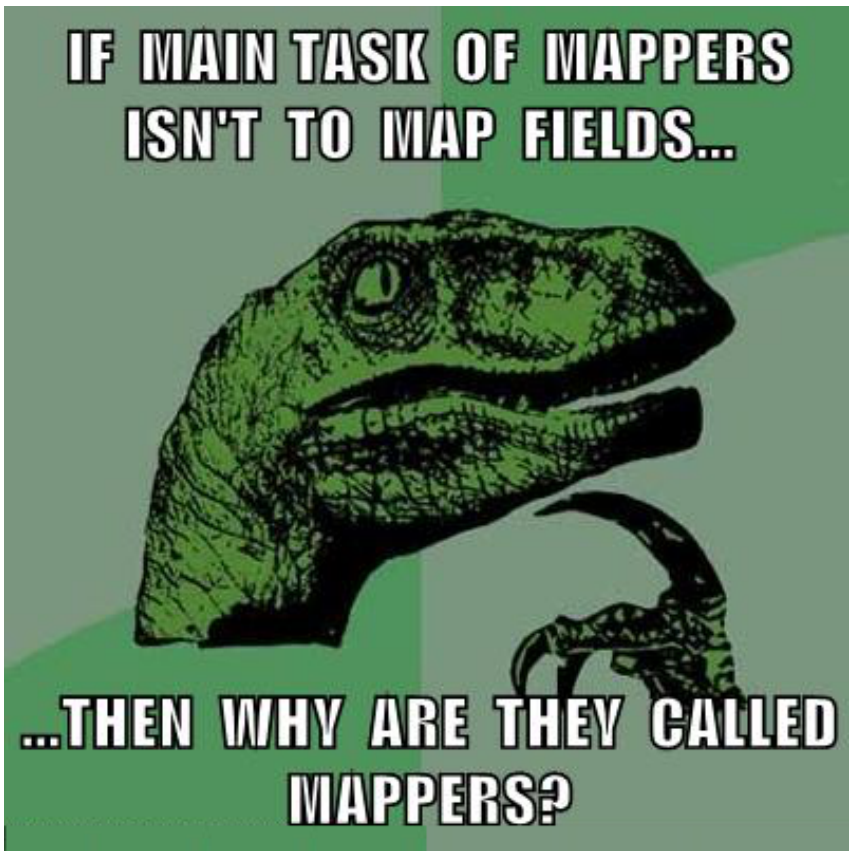
Scala related features

- Implicits of all kinds
- Point-free style
- Scala virtualized
- Macros

Yajom

- Yet Another Java Object Mapper
- Based on implicits and Scala macros
- All reflection is done in compile-time
- Easy to configure, extend or derive
  (having no clue about Scala Macros at all)

Philosoraptor



Java with JAXB plugin

```
1  A a = from.getA();
2  if (verySpecial(from)) {
3    B b = defaultB;
4    if (a != null) {
5      B realB = a.getB();
6      if (realB != null)
7        b = realB;
8    }
9    to.c().d().e().setB(b)
10 }
```

Option-based yajom

```
1  yajomOption(to.getC.getD.getE.setB) {
2    val a = from.getA
3    if (verySpecial(from)) {
4      var b = defaultB
5      if (a != null) {
6        val realB = a.getB
7        if (realB != null)
8          b = realB
9      }
10     Some(b)
11   } else
12     None
13 }
```

Option-based yajom with maybe

```
1 yajomOption(to.getC.getD.getE.setB) {
2   if (verySpecial(from)) {
3     val b = maybe(from.getA.getB)
4     Some(b.getOrElse(defaultB))
5   } else
6     None
7 }
```

Yajom

Talk is cheap. Show me the code.

Yajom Pros

- Scala

Yajom Pros

- Scala
- Compile-time bulletproof

Yajom Pros

- Scala
- Compile-time bulletproof
- IDE support

Yajom Pros

- Scala
- Compile-time bulletproof
- IDE support
- Works with any classes without any changes

Yajom Pros

- Scala
- Compile-time bulletproof
- IDE support
- Works with any classes without any changes
- No problem with debugging

Yajom Cons

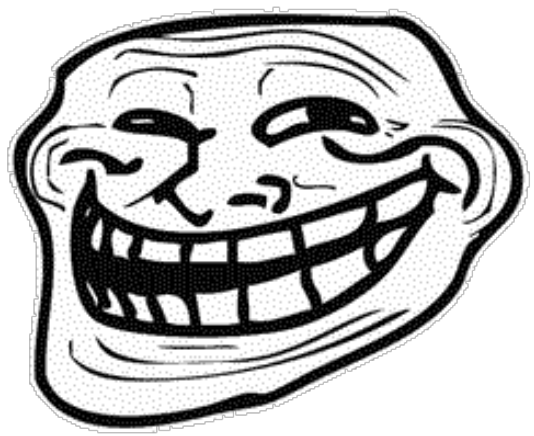
- Scala

Yajom Cons

- Scala
- Maybe not mature enough

Yajom Cons

- Scala
- Maybe not mature enough



Yajom

https://github.com/gark87/yajom

Thank You!